Algorithms Comparison for Stutter Detection and Classification

Ramitha V[1], Rhea Chainani[1], Saharsh Mehrotra[1], Sakshi Sah[1], and Dr.Smita Mahajan[1]

[1]Symbiosis Institute of Technology, Pune

Abstract

Stuttering is a neuro-developmental speech disorder that interrupts the flow of speech due to involuntary pauses and sound repetitions. It has profound psychological impacts that affect social interactions and professional advancements. Automatically detecting stuttering events in speech recordings could assist speech therapists or speech pathologists track the fluency of people who stutter (PWS). It will also assist in the improvement of the existing speech recognition system for PWS. In this paper, the SEP-28k dataset is utilized to perform comparative analysis to assess the performance of various machine learning models in classifying the five dysfluency types namely Prolongation, Interjection, Word Repetition, Sound Repetition and Blocks.

Keywords: Stuttering, Speech Disorder, Automatic Dysfluency detection, Machine Learning, Comparative Analysis.

1. Introduction

Stuttering is a type of speech disorder which is characterized by interruptions in speech or recurrence of words, phrases or sounds. Stuttering generally results in disruption of normal flow of speech and is often accompanied by the tremor of lips or rapid eye blinks. Stuttering is a neuro-developmental speech disorder that occurs due to the disruptions of the neurological connections [**1**].

Stuttering affects 1% of the world's population and it has impacted around 5% to 17% of the population at some point in their lives. Stuttering often resolves naturally or due to clinical intervention, mostly before the age of 6. It is more prevalent in younger children, with a decrease in cases as age increases. The recovery rate is approximately 50% in older children after receiving therapy. However, recovery rates vary and stuttering can

persist in adulthood and can lead to chronic, persistent stuttering with some individuals experiencing long-term effects. Therefore, about 70 million individuals suffer from this difficulty [**2**]. Anxiety disorders are significantly more prevalent in stuttering children than non-stuttering children, with the former being six times more likely to develop clinical social anxiety disorders than the latter [**3**]. Stuttering also interferes with work, school, and family life, and is a reason for embarrassment for PWS leading them to avoid situations where they are required to speak. 4 out of 10 adults have reported that they were refused job, promotion, and educational opportunities due to their stuttering [**4**]. Due to these reasons, PWS develop social anxiety, fear, shame, etc. that negatively impact their quality of life.

A detailed review of the past two decades of research reveals a concerted effort to develop automated methods for identifying stuttering. Researchers have meticulously examined datasets, acoustic features and classification methodologies to elucidate the challenges and opportunities in this domain. Key among these efforts is the extraction of acoustic features, where methods ranging from autocorrelation to spectral analysis have been explored. Recent advancements have seen the integration of convolutional neural networks (CNNs) with spectrogram representations, showcasing the potential for enhanced accuracy in stuttering identification tasks. The research meticulously outlines the various statistical machine learning techniques employed for stuttering detection and classification. In most cases, Support vector machines (SVM), artificial neural networks (ANNs) and hidden Markov models (HMM) have been proved to be most effective. SVMs have emerged as the most prevalent classifier, demonstrating high accuracy across different types of stuttering. Methods including k-nearest neighbor (k-NN) and linear discriminant analysis (LDA) have shown further promise, enriching the vast scope of stutter identification methodologies. However, the past research also underscores a paradigm shift towards deep learning techniques, particularly CNNs and recurrent neural networks (RNNs), in stuttering identification. These approaches offer automatic feature extraction capabilities and have exhibited superior performance compared to traditional methods. Recent studies, including the introduction of models like FluentNet and StutterNet, have addressed data scarcity concerns and achieved remarkable accuracies on larger datasets. It is noteworthy that the accuracies achieved by these models are indeed impressive, with SVMs reporting high accuracies ranging from 75% to 98% across various stuttering types[**5**]. Similarly, CNN models have displayed average accuracies of 91.15% to 91.75% on different datasets[**6**]. While challenges persist in generalizing models to larger datasets and capturing diverse stuttering patterns effectively, the advancements in machine learning and deep learning offer renewed hope for the automated identification of stuttering.

The study of Stuttering Detection lies within the domain of speech and language processing. The main focus of the research is the identification and classification of the speech

dysfluencies vis-a-vis stuttering. This paper essentially uses signal processing techniques to analyze the acoustic properties of speech. This includes extracting key features from speech signals such as the spectral features in order to develop high-level models which are capable of monitoring stuttered speech patterns.

The traditional method of speech therapists or pathologists analyzing the speech or the recordings of PWS manually to assess the type and severity of stuttering [**7**] is time-consuming and intense and calls for an automated system for the task [**1**].

The methodology proposed through this paper aims to contribute to this area of research in the following manner:

1. Developing robust machine learning models for classifying the five classes of Stutter - Interjection, Prolongation, Blocks, Sound Repetitions and Word Repetitions.
2. Conducting analysis on the impact of various features extracted manually as well as using pre-trained models.
3. Performing comparative analysis on each class of stutter using various machine learning models.

Further in this paper, Section 2 consists of a detailed review of existing literature on stutter detection and classification methods, including recent advancements, challenges and opportunities. Section 3 discusses the data used, while Section 4 and Section 5 elucidate the proposed methodology. Section 6 presents the results of the experiments, including accuracy scores and comparison with existing literature. Lastly, Section 7 concludes the paper by summarizing key findings, discussing limitations and suggesting future research directions.

2. Literature Review

Recent developments showcase an increasing use of pre-trained models such as Whisper and wav2vec for feature extraction. These features are further utilised by models such as SVM and LSTM to classify the different classes of stutter in datasets such as SEP-28k, FluencyBank and KSOF. Other methods such as data augmentation and multi-contextual deep learning have been employed to improve the performance of the models. Despite all the recent developments there are certain issues such as generalizability, cross-language transferability, biases and class imbalances.

Table 1: Literature Review Table for Stutter Detection

Ref no	**Dataset**	**Model Used**	**Performance**	**Advantage**	**Drawback**
[**8**]	SEP-28k,KSOF	Fine tuning a pre-trained wav2vec 2.0, multi-class learning and Support Vector Machine	Best F1 score for Fluency Bank is 0.60 and KSOF is 0.76	Cross-lingual transferability, Robustness and versatility	Limited scope dysfluencies, Bias in data and Speaker variability
[**9**]	SEP-28k-E,FluencyBank, KSoF	Modified wav2vec 2.0 system	Best F1-score is 0.80 for Modified	Data quantity and Diversity	Generalisation
[**10**]	SEP-28K	wav2vec 2.0, ECAPA-TDNN and LDA	Best accuracy is 68.35 %	wav2vec 2.0 performance and contextual embeddings	Generalisation and fine-tuning
[**11**]	SEP-28K	LSTM	Weighted accuracy is 83.6% and the F1 score is 83.6%	Improved Performance spectral features with pitch	Prolongation and Blocks are difficult to detect
[**12**]	KSoF,SEP-28k	wav2vec 2.0,LSTM and SVM	Highest mean(std) per metric is 0.73(0.05) for modified	New dataset and cross language transferability	Less satisfactory results for LSTM and word repetition not detected well.
[**13**]	SEP-28K,MUSAN, UCLASS, FluencyBank, LibriStutter	DNN, ConvLSTM, ResNet BiLSTM	F1-score(%) is 44.07	Handling class imbalance and data augmentation	Doesn't perform well on real-life data and cannot predict exact time stamp occurence of stuttering.
[**14**]	SEP-28k extended	SVM on wav2vec 2.0 features	Best f1-score is 0.73 for interjections	Generation of semi-automatic labels and publishing extended dataset	Not using class specific methods for classification.
[**15**]	SEP-28k, FluencyBank	Feature extraction using Whisper and SVM	Best averaged f1-score is 0.81	Reduced number of learning parameters that increased efficiency	Performance degradation, model's runtime and generalizability.
[**16**]	UCLASS, LibriStutter	LSTM	Average accuracy is 86.90	Validation on synthetic datasets	Bias and generalizability

3. Data and Description

3.1 Stuttering Events in Podcasts (SEP-28k)

SEP-28k is a publicly available dataset published by Apple Machine Learning Research in 2021. The extraction of the audio clips comprises of two steps. First step involves extraction of the clips using the *SEP-28k_ episodes* CSV provided in the GitHub repository [**20**]. After the extraction of the audio files, the next step is to segment the audio files into 3 second clips. The dataset originally consisted of audio files from 8 podcast shows but 2 of the podcasts (namely, StutteringIsCool and StrongVoices) are no longer available, thus the dataset finally consisted of 6 shows i.e., 265 episodes instead of 385 [**17**].

3.1.1 Annotations

Annotations typically are the labels associated with the audio clips. SEP-28k dataset was labeled by atleast three annotators. The annotations in the dataset include labels such as 'Unsure', 'PoorAudioQuality', 'Music', 'DifficultToUnderstand', 'Interjection', 'Prolongation', 'Blocks', 'WordRep', 'SoundRep', 'NoStutteredWords', 'NoSpeech' and 'NaturalPause', all of which are explained in Table 2. The annotations were done on the basis of a time-interval based assessment. The 3 second audio clips are annotated with binary labels. A clip may contain multiple dysfluency types, thus making the SEP-28k dataset multi-label and multi-class. The inter-annotator agreement as well as disagreement states:

- *NoStutteredWords* and the 5 dysfluency classes have negative correlation - indicating that the annotators do not label a sample with those classes together hence depicting inter-annotator agreement.
- *Block* and *Natural Pause* have weak positive correlation (0.11) indicating that the annotators have often had a hard time differentiating between the two (as those are often differentiable only based on physical signs).
- *PoorAudioQuality* and *DifficultToUnderstand* also have weak positive correlation (0.20) indicating that annotators often label same audio samples with either of those two classes.

The inter-annotator agreement also states that word repetitions, sound repetitions, interjections and no dysfluencies were more consistent and that blocks and prolongations had only slight agreement indicating that blocks are harder to detect just based on the audio clips [**11**].

Table 2: Stuttering Labels [**11**]

Stuttering Labels	**Definition**	**SEP-28k (%)**
Block	Gasps for air or stuttered pauses	12.0
Prolongation	Elongated syllable (e.g., *M[mmm]ommy*)	10.0
Sound Repetition	Repeated syllables (e.g., I [*pr-pr-pr-]prepared dinner*)	8.3
Word/Phrase Repetition	The same word or phrase is repeated (e.g., *I made [made] dinner*)	9.8
No dysfluencies	Confirmation that none of the above is true.	56.9
Interjection	Common filler words such as "*um*" or "*uh*" or person-specific filler words that individuals use to cope with their stutter (e.g., some users frequently say "*you know*" as a filler).	21.2
Non-dysfluent labels		
Natural Pause	There is a pause in speech that is not considered a block or other disfluency.	8.5
Difficult To Understand	It is difficult to understand the speech.	3.7
Unsure	An annotator selects this if they are not confident in their labeling.	0.1
No Speech	There is no speech in this clip. It is either silent or there is just background noise.	1.1
Poor Audio Quality	It is difficult to understand due to, for example, microphone quality.	2.1
Music	There is background music playing.	1.1

4. Proposed Work Diagram

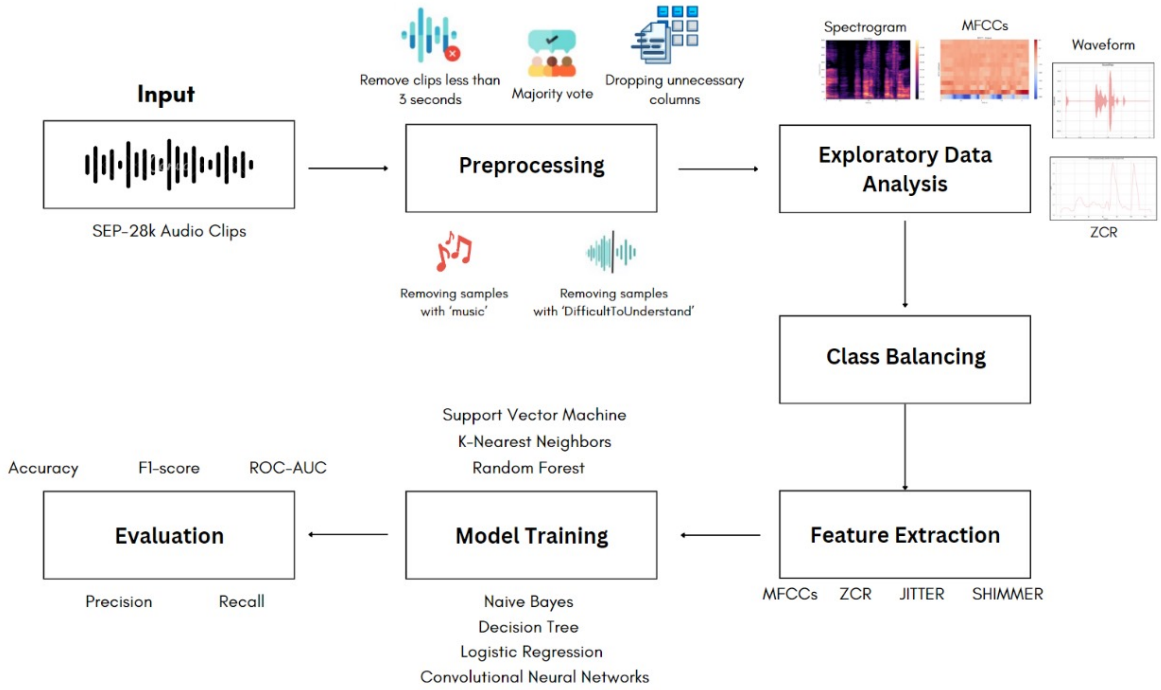

Figure 1: Methodology Diagram

5. Proposed Methodology

5.1 Dataset Acquisition

In the methodology proposed, SEP-28k dataset, which is publicly available in the Apple Machine Learning Research, has been used. Following the extraction of the clips in .mp3 format, a conversion process was employed to convert the .mp3 files into .wav format. Subsequently, 3-second clips were extracted from the audio files for each podcast episode.

5.2 Preprocessing

The initial step of preprocessing was deleting the audio clips which were not 3 seconds long and also to check if all the audio clips have a sampling rate of 16kHz. Additionally, significant preprocessing was required due to the disparities between the annotators. Initially the dataset consisted of 11 classes which made it multi-class and multi-label. Several of the classes were dropped due to their lack of any meaningful contribution to the model. Firstly, the *Unsure* column is dropped. Moving forward, the audio clips labeled as *music*, which had introductory or concluding music from the podcast, were also dropped, following which the *music* column was also dropped. Furthermore, for the *DifficultToUnderstand* column, the audio samples where two or more annotators agreed that the audio clip was difficult to understand were removed and then the *DifficultToUnderstand* column was also removed from the dataset.

5.2.1 Handling Class Imbalance

The SEP-28k data suffers from severe class imbalance. The *NoStutteredWords* class is overrepresented compared to the others. As can be seen in Figure 2, the count of samples in the *NoStutteredWords* class alone equals the total count of samples in all the other classes combined. Within each of the individual classes also, class imbalance can be noticed. The negative class has a significantly greater number of samples than the positive class. Class imbalance is one of the most important problems that has to be addressed prior to training a model. Class imbalance results in the underrepresentation of the minority class where the learning is heavily dominated by the majority class. In some scenarios, minority class samples are not detected by the model and hence results in the poor performance of the chosen models.

The proposed methodology addresses the class imbalance problem by considering each of the 5 classes - interjection, prolongation, word repetition, sound repetition and blocks as individual binary classifications. In each of the classes, undersampling was used i.e., after identifying the minority class, a subset of the majority class is randomly sampled to match the number of samples in the minority class. By sampling them in the method mentioned above, it is ensured that both the negative and the positive class samples are equally represented.

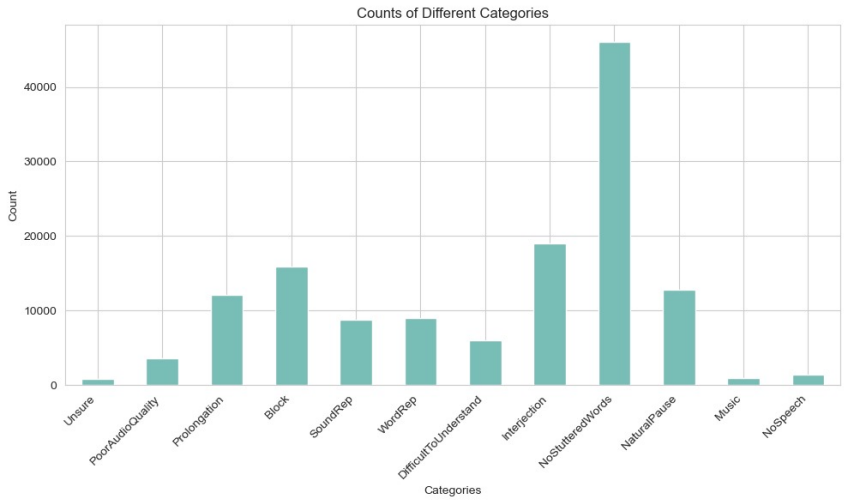

Figure 2: Count of categories

5.3 Feature Extraction

Features are a crucial part of any machine learning algorithm. Hence it is crucial to choose the desired features. The primary features considered are Mel-Frequency Cepstral Coefficients(MFCCs), Zero Crossing Rate, Jitter and Shimmer. Other sets of features that were considered are Pitch, Energy and Perceptual Linear Prediction(PLP)[**18**]. These features were not considered as the combination of these features with the primary features brought down the accuracy of the models. Experimentation was also done with the pretrained wav2vec 2.0 model for feature extraction. The various features [**19**] considered are:

5.3.1 MFCCs

Mel-frequency cepstral coefficients or MFCCs represent the spectral characteristics extracted from the audio clips. The calculation of MFCCs involve conversion of audio clips into the frequency domain using Fourier transform (FT) and then mapping the spectrum to the mel scale. Further, logarithm of the mel spectrum is calculated and then Discrete Fourier Transform (DCT) is calculated to reduce the dimensionality.

5.3.2 Zero Crossing Rate

Zero Crossing Rate or ZCR represents the measure at which the audio signal changes its sign. ZCR is interpreted as the measure of noisyness of a signal. A high ZCR implies a rapid change in the waveform which is associated with a high frequency whereas a low ZCR implies steady or sustained changes.

5.3.3 Jitter

Jitter represents the variation of pitch periods in the consecutive cycles. Jitter typically represents the irregularity in the fundamental frequency (pitch) over time. Higher Jitter

values indicate a greater variability in the fundamental frequency and corresponds to the presence of vocal disorders.

5.3.4 Shimmer

Shimmer represents the variation of amplitude or intensity of pitch periods in the consecutive cycles. Shimmer typically represents the irregularity in the intensity/loudness over time. Higher shimmer values indicate a greater variability in the intensity which also implies the presence of vocal disorders.

5.4 Model Training

After the extracted features are concatenated into an array, these features are given to the models for training. Here individual models have been considered for each class - namely, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Random Forest, Decision Tree and Naive Bayes. All the above mentioned models are trained for all the classes and the model which gives the highest accuracy is chosen for that particular class.

5.5 Hyperparameter Tuning

Hyperparameter tuning and cross validation techniques were applied on each of the models i.e., KNN, Logistic Regression, Decision Tree, Random Forest, and SVM, and just cross validation (with 10-fold cross-validation) for Naïve Bayes as it is a non-parametric model.

5.6 Evaluation Metrics

Both training and testing were done on the SEP-28k dataset where the train to test ratio is 70:30. The main evaluation metric considered for choosing the models for each class is accuracy, as the data for each class was balanced. Other metrics considered are F1-score, accuracy, precision, recall, confusion matrix and ROC-AUC.

6. Results

After comparing all the models for each of the dysfluency classes, it can be inferred from Table 3 and Figure 3 that the best accuracy was achieved using Random Forest for Prolongation (0.66), KNN for Sound Repetitions (0.63) and Blocks (0.60), and SVM for Word Repetition (0.63) and Interjection (0.67).

Table 3: Accuracy scores for different models

	Prolongation	**SoundRep**	**WordRep**	**Interjection**	**Block**
RandomForest	**0.66**	0.63	0.61	0.65	0.58
SVM	0.63	0.63	**0.63**	**0.67**	0.60
KNN	0.60	**0.63**	0.60	0.64	**0.60**
Naïve Bayes	0.60	0.57	0.56	0.58	0.51
Logistic Regression	0.55	0.56	0.56	0.60	0.54
Decision Tree	0.58	0.58	0.55	0.57	0.56
CNN	0.60	0.62	0.56	0.61	0.56

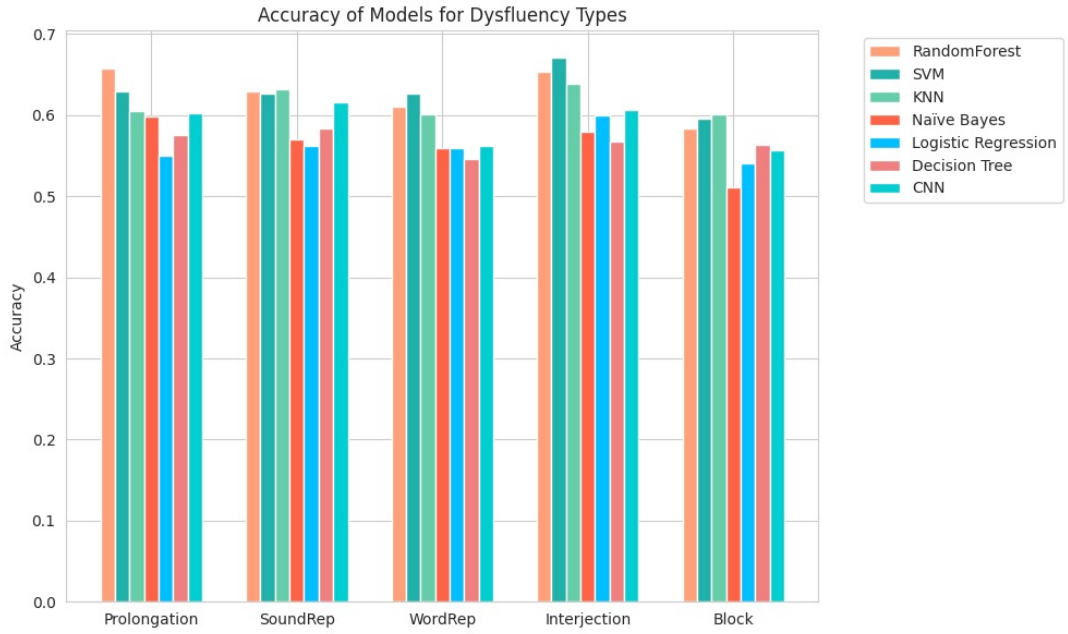

Figure 3: Comparision of Accuracy of different models for all the dysfluencies

From Figure 4 it can observed that the Word Repetitions class shows the most balanced performance with approximately equal counts in both correct identifications (True positives: 443, True negatives: 444) and errors (False Positives: 264, False Negatives: 264). Following closely is the Interjection class, which also exhibits a nearly equal distribution of correct identifications and errors (True positives: 951, True negatives: 964, False Positives: 487, False Negatives: 500). The model used for both of these dysfluencies was SVC, suggesting its effectiveness in achieving such balance compared to other models. The remaining classes show moderate performance with varying degrees of imbalance between correct identifications and errors.

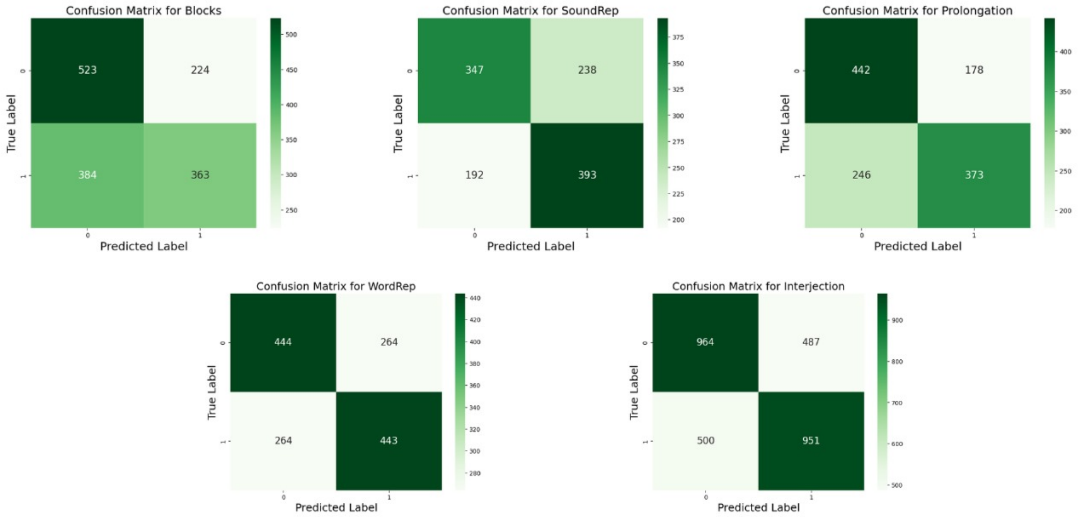

Figure 4: Confusion Matrix of the best model for each of the dysfluencies

The ROC AUC values from Figure 5 indicate variations in discriminatory performance among classifiers across dysfluency types. Among the highest accuracy models chosen for each class, the Support Vector Classifier (SVC) generally performs well, particularly evident in its higher ROC AUC scores for interjections (0.73) and word repetitions (0.68). In comparison, Random Forest and KNN demonstrate moderate performance, with ROC AUC scores ranging from 0.66 to 0.71 across different dysfluencies. Notably, SVC consistently outperforms other models in discrimination ability across dysfluency types, suggesting its effectiveness in classification tasks for stuttering events.

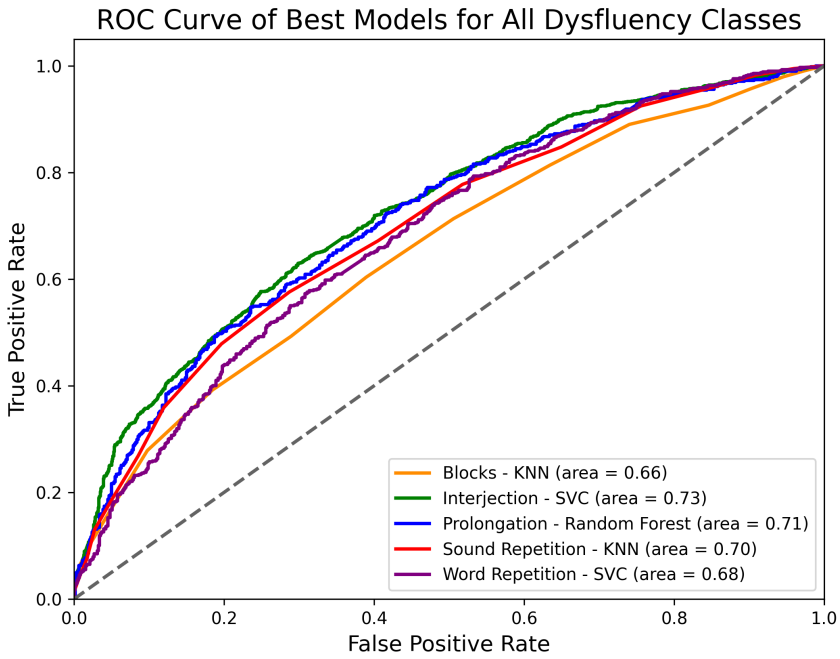

Figure 5: ROC Curve of the best model for each of the dysfluencies

In contrast to [**14**], which only employs SVM for every class, the proposed work here implements different models for each dysfluency class. From Table 4 it can be inferred that the proposed model outperforms the models used in [**14**] for three of the classes - Blocks (0.55), Prolongations (0.64) and Word Repetitions (0.63) in terms of F1-score.

Table 4: Comparison with existing literature

	ML Model Selected	**F1-score**	**F1-score from [14]**
Blocks	KNN	**0.55**	0.40
Interjection	SVM	0.67	0.73
Prolongation	Random Forest	**0.64**	0.54
Sound Repetition	KNN	0.65	0.70
Word Repetition	SVM	**0.63**	0.50

7. Conclusion

In conclusion, this study presents a comprehensive approach to automated stutter detection and classification in audio files, utilizing the SEP-28k dataset and comparing various machine learning models. The proposed models exhibited superior performance compared to existing literature[**14**] in specific stuttering events, namely, Blocks, Prolongation and Interjections.

Although there were limitations such as the inter-annotator disagreement, class imbalance and the diversity of stuttering patterns in the data, they were effectively resolved by the proposed methodology.

Future scopes include exploring alternative feature sets and deep learning architectures to further enhance accuracy, and correction of dysfluencies based on speech samples which fosters accessibility in everyday communication among PWS and smart voice assistants.

References

[1] Sheikh, Shakeel A., Md Sahidullah, Fabrice Hirsch, and Slim Ouni. "Machine learning for stuttering identification: Review, challenges and future directions." Neurocomputing (2022)

[2] E. Yairi, N. Ambrose, Epidemiology of stuttering: 21st century advances, Journal of Fluency Disorders 38 (2) (2013) 6687.

[3] L. Iverach, M. Jones, L. F. McLellan, H. J. Lyneham, R. G. Menzies, M. Onslow, R. M. Rapee, Prevalence of anxiety disorders among children who stutter, Journal of Fluency Disorders 49 (2016) 1328

[4] N. S. A. NSA, The experience of people who stutter: A survey by the national stuttering association, New York, NY: Author.

[5] J. P´alfy, J. Posp´ıchal, Recognition of repetitions using support vector machines, in: Signal Processing Algorithms, Architectures, Arrangements, and Applications SPA 2011, IEEE, 2011, pp.

[6] 1–6.T. Kourkounakis, A. Hajavi, A. Etemad, Detecting multiple speech disfluencies using a deep residual network with bidirectional long short-term memory, in: Proc. ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, pp. 6089–6093.

[7] B. Guitar, Stuttering: An Integrated Approach to its Nature and Treatment, Lippincott Williams & Wilkins, 2013.

[8] Bayerl, Sebastian P., Dominik Wagner, Elmar Nöth, and Korbinian Riedhammer. "Detecting dysfluencies in stuttering therapy using wav2vec 2.0." arXiv preprint arXiv:2204.03417 (2022)

[9] Sebastian P. Bayerl, Dominik Wagner, Florian Hönig, Tobias Bocklet, Elmar Nöth, & Korbinian Riedhammer. (2022). Dysfluencies Seldom Come Alone – Detection as a Multi-Label Problem.

[10] Sheikh, S. A., M. Sahidullah, F. Hirsch, and S. Ouni. "Introducing ECAPA-TDNN and Wav2Vec2. 0 embeddings to stuttering detection. arXiv 2022." arXiv preprint arXiv:2204.01564.

[11] Lea, Colin, Vikramjit Mitra, Aparna Joshi, Sachin Kajarekar, and Jeffrey P. Bigham. "Sep-28k: A dataset for stuttering event detection from podcasts with people who stutter." In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6798-6802. IEEE, 2021

[12] Sebastian P. Bayerl, Alexander Wolff von Gudenberg, Florian Hönig, Elmar Nöth, & Korbinian Riedhammer. (2022). KSoF: The Kassel State of Fluency Dataset – A Therapy Centered Dataset of Stuttering.

[13] Shakeel A. Sheikh, Md Sahidullah, Fabrice Hirsch, & Slim Ouni. (2023). Advancing Stuttering Detection via Data Augmentation, Class-Balanced Loss and Multi-Contextual Deep Learning.

[14] Bayerl, Sebastian P., Dominik Wagner, Elmar Nöth, Tobias Bocklet, and Korbinian Riedhammer. "The influence of dataset partitioning on dysfluency detection systems." In International Conference on Text, Speech, and Dialogue, pp. 423-436. Cham: Springer International Publishing, 2022

[15] Ameer, Huma, Seemab Latif, Rabia Latif, and Sana Mukhtar. "Whisper in Focus: Enhancing Stuttered Speech Classification with Encoder Layer Optimization." arXiv preprint arXiv:2311.05203 (2023)

[16] Kourkounakis, Tedd, Amirhossein Hajavi, and Ali Etemad. "FluentNet: end-to-end detection of speech disfluency with deep learning." arXiv preprint arXiv:2009.11394 (2020)

[17] Lea, Colin, Vikramjit Mitra, Aparna Joshi, Sachin Kajarekar, and Jeffrey P. Bigham. "Sep-28k: A dataset for stuttering event detection from podcasts with people who stutter." In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6798-6802. IEEE, 2021.

[18] Sheikh, Shakeel A., Md Sahidullah, Fabrice Hirsch, and Slim Ouni. "Machine learning for stuttering identification: Review, challenges and future directions." Neurocomputing 514 (2022): 385-402.

[19] McKinney, Martin, and Jeroen Breebaart. "Features for audio and music classification." (2003).

[20] Alnashwan, Raghad, Noura Alhakbani, Abeer Al-Nafjan, Abdulaziz Almudhi, and Waleed Al-Nuwaiser. "Computational Intelligence-Based Stuttering Detection: A Systematic Review." Diagnostics 13, no. 23 (2023): 3537